Intelligent Automation & Soft Computing
DOI:10.32604/iasc.2022.022438
Article

Tech Science Press

Fair and Stable Matching Virtual Machine Resource Allocation Method

Liang Dai[1], AoSong He[1], Guang Sun[1,3] and Yuxing Pan[2,*]

[1]Hunan University of Finance and Economics, Changsha, 410205, China
[2]Changsha University of Science and Technology, Changsha, 410114, China
[3]University of Alabama, Tuscaloosa, 35401, USA
*Corresponding Author: Yuxing Pan. Email: simon5115@163.com
Received: 07 August 2021; Accepted: 26 September 2021

Abstract: In order to unify the management and scheduling of cloud resources, cloud platforms use virtualization technology to re-integrate multiple computing resources in the cloud and build virtual units on physical machines to achieve dynamic provisioning of resources by configuring virtual units of various sizes. Therefore, how to reasonably determine the mapping relationship between virtual units and physical machines is an important research topic for cloud resource scheduling. In this paper, we propose a fair cloud virtual machine resource allocation method of using the stable matching theory. Our allocation method considers the allocation of resources from both user's demand and cloud computing resource provider's request. When multiple users apply for resources, firstly select a user by user priority, and then deal with this user's task. Because the user priority is dynamic, so as to avoid a user's long-term share of resources. This strategy makes user task scheduling is relatively fair. On the basis of weighing the fair allocation of user resources, the stable matching between physical machines and virtual machines is achieved. Our simulation experiments especially given that the main focus of the paper is not to develop a very novel algorithm, but to demonstrate our virtual machine resource allocation method, which effectively improves the average utilization rate of computing resources and reduces the operating costs of cloud providers.

Keywords: Cloud computing; stable matching; resource allocation; user fairness

1 Introduction

The continuous development of the Internet has led to a rapid increase in the number of netizens, increasingly diversified computing needs, and more and more computing resources. But for some companies, it is very uneconomical to spend huge sums of computing resources to meet temporary needs. If the computing resources can be leased to bring economic benefits to themselves while fulfilling their own computing needs, then the idle resources will realize its value and bring economic benefits to the enterprise. It also allows the rented users to spend relatively little money to obtain computing resources.

Cloud computing was born under this idea. Its approach is to virtualize a large number of computing resources in the data center (including resources such as networks, servers, storage, application software, services, etc.) into a resource pool, and realize effective use of resources through middleware. And integration, allocating to users on demand. However, due to the massive amount of resources, how middleware allocates resources to meet user needs, while reducing supplier costs to the greatest extent, and maximizing the reasonable allocation of resources is still a focus in the field of cloud computing research at home and abroad.

Cloud computing has a wide range of applications. For example, virtualization technology can be used to build public platform server clusters in the field of e-government, and Platform As A Service (PAAS) technology can be used to build public service systems. In the medical field, it can be used for DNA information analysis, massive case storage analysis, and medical image processing. In the field of transportation, it can be used to identify unlicensed vehicles; in the field of scientific research, it can be used for earthquake monitoring and marine information monitoring. With the widespread use of smart phones and mobile offices, cloud computing is quietly changing people's daily lives.

Users and suppliers are the main participants in the cloud system. From an economic point of view, the problem of resource allocation is how to maximize the benefits of users and suppliers. The overall limited resources are allocated to users to realize user needs and improve user satisfaction. Make full use of resources to ensure the economic benefits of suppliers.

Currently, in the literature on resource allocation issues, most allocation methods are proposed on the basis of considering the interests of one of the parties. However, as two different entities, the user and the supplier have different pursuit of interests, and the transaction can proceed normally only if their respective interests are guaranteed, otherwise the supplier will refuse to provide resources to the user, and the user will also Choose other suppliers that are more beneficial to you. Therefore, in this emerging business service model, the resource allocation method needs to consider the interests of users and suppliers at the same time, so that cloud computing can continue to develop.

2 Related Works

Resource allocation has always been one of the core issues of cloud computing [1]. The goal of resource allocation is to allocate the smallest resources to consumers, while providing maximum satisfaction and maximizing benefits. In the process of resource allocation, it is most important to choose a suitable resource allocation algorithm. In a specified resource environment, different resource usage rules will result in different resource usage, and may even affect the performance of the entire cloud system [2]. Research on cloud computing resource allocation methods has been carried out at home and abroad. The main research directions at present are: research on resource allocation and scheduling with the goal of reducing energy consumption, research on resource allocation and scheduling with the goal of improving utilization, and cloud resources based on economics. Research on management model and on-demand dynamic configuration method of cluster resources [3].

Shi Xuelin and Xu Ke proposed a cloud utility maximization model by referring to the network utility maximization model [4]. Compared with the traditional scheduling model, the objective function is no longer to minimize the maximum completion time, but to achieve the maximum utility as the scheduling goal, which can fully improve user satisfaction. However, its computing power measurement method only considers the pure CPU and does not expand to the common constraints of multiple resources; nor does it explore the impact of different utility functions on the efficiency of virtual machine resource scheduling.

Aiming at the energy consumption of cloud computing data centers, Luo Liang, Wu Wenjun, and Zhang Fei proposed a highly accurate energy consumption model to predict the energy consumption of a single server in the cloud computing data center [5]. They analyze and summarize the influence of different parameters and methods on server energy consumption modeling, and proposed a server energy consumption model suitable for cloud computing data center infrastructure. Nguyen Minh, Nhut Pham, Van Son Le and others task reducing the number of physical machines that provide resources for virtual services in cloud computing is one of the effective ways to reduce energy consumption [6].

In terms of cloud computing resource scheduling, CPU and memory resources are generally used as constraints or the application scheduling virtual machine and the allocation of physical computing resources to the virtual machine are modeled into a constraint satisfaction problem model, which optimizes the allocation of virtual machine resources and improves resource utilization. . Chen Xiaojiao, Chen Shiping and others proposed a group-based multi-objective genetic algorithm virtual machine resource allocation algorithm [7]. Through the improved genetic algorithm, the combination coding and resource requirement coding of the virtual machine are carried out, and the number of physical machines and the physical machine resources occupied by the virtual machines are integrated. The experimental results show that in the process of realizing the matching between physical machines and virtual machines, the algorithm is effective for reducing the number of physical machines used, improving resource utilization, and achieving the purpose of energy saving. But no further research has been done on the correlation between virtual machines.

Seyedeh Aso Tafsiri, Saleh Yousefi studied a combined double auction-based market [8], in which a broker performs the allocation of provider virtual machines according to user requests. The proposed allocation problem is expressed as an integer linear programming model, which aims to maximize the total profit of users and providers. The literature proposed a cloud computing resource optimization allocation strategy based on the game evolution strategy to address the problem of the market resource needs to be allocated on demand and satisfy the rationality and fairness in cloud computing [9]. This strategy uses genetic and evolutionary algorithms to meet the needs of the rationality and fairness of resource allocation from a macro perspective, thus solving the problem that traditional methods only consider individual characteristics.

Because different auction models provide various market-driven resource allocation mechanisms. Literature studies cloud computing resource allocation based on auctions [10]. The article gives an auction-based cloud computing resource allocation framework, discusses the main problems that need to be faced when designing a dynamic resource allocation mechanism, and shares the latest technology of auction-based cloud computing resource allocation.

In solving the problem of dynamic resource allocation, the literature proposed Service based system (SBS) to apply dynamic resource allocation method [11]. When user demand changes dynamically, the application load in SBS will be different at different moments. In order to cope with this change, this method combines the initial static resource allocation scheme with the dynamic resource allocation scheme, and requires the application throughput constraints of SBS. The number of resources required by the application is calculated to calculate the resource allocation time of the application to ensure the end-to-end performance of the SBS application and improve resource utilization.

In a cloud system, due to the continuous changes in requirements and environments, the types and numbers of virtual machines running on nodes need to be continuously adjusted according to requirements. Therefore, Mi Boer and Wang Huaimin proposed a dynamic configuration method of cluster resources [12]. This method is based on the idea of genetic algorithm in the resource allocation under the premise of meeting the needs of users, through the use of chromosome coding, to realize the rapid and dynamic allocation of resources. The Boolean quadratic exponential smoothing method is used to predict user requests, thereby avoiding the reconfiguration result later than the demand change and improving the utilization of cluster resources. However, this method only considers the dynamic allocation of resources within a single cluster. When considering the dynamic allocation of resources across clusters, how to effectively resolve decision conflicts in multiple front-end environments requires further research.

Matching problems are born with the emergence of decision-making, whether in terms of algorithms or in our daily lives, the stability of matching is very important for decision-making. Since David Gale and Lloyd Shapley first proposed the stable matching theory in 1962, the "stability" of matching has received a lot of attention, and research on this topic has also Has been very active. Economists Shapley and Ross Rothy sought a stable solution to the bilateral matching problem from the perspective of mathematics and games, and proposed a game-based stable matching theory research method [13]. In 2012, they won the Nobel Economy by virtue of this theory.

Resource allocation has always been one of the core issues of cloud computing [1]. The goal of resource allocation is to allocate the smallest resources to consumers, while providing maximum satisfaction and maximizing benefits. In the process of resource allocation, it is most important to choose a suitable resource allocation algorithm. In a specified resource environment, different resource usage rules will result in different resource usage, and may even affect the performance of the entire cloud system [2]. Research on cloud computing resource allocation methods has been carried out at home and abroad. The main research directions at present are: research on resource allocation and scheduling with the goal of reducing energy consumption, research on resource allocation and scheduling with the goal of improving utilization, and cloud resources based on economics. Research on management model and on-demand dynamic configuration method of cluster resources [3].

Shi Xuelin and Xu Ke proposed a cloud utility maximization model by referring to the network utility maximization model [4]. Compared with the traditional scheduling model, the objective function is no longer to minimize the maximum completion time, but to achieve the maximum utility as the scheduling goal, which can fully improve user satisfaction. However, its computing power measurement method only considers the pure CPU and does not expand to the common constraints of multiple resources; nor does it explore the impact of different utility functions on the efficiency of virtual machine resource scheduling.

Aiming at the energy consumption of cloud computing data centers, Luo Liang, Wu Wenjun, and Zhang Fei proposed a highly accurate energy consumption model to predict the energy consumption of a single server in the cloud computing data center [5]. They analyze and summarize the influence of different parameters and methods on server energy consumption modeling, and proposed a server energy consumption model suitable for cloud computing data center infrastructure. Nguyen Minh, Nhut Pham, Van Son Le and others task reducing the number of physical machines that provide resources for virtual services in cloud computing is one of the effective ways to reduce energy consumption [6]. A resource allocation problem to reduce energy consumption is proposed. The ECRA-SA algorithm is designed for this problem and the meta-heuristic algorithm is applied to estimate the result of the problem.

In terms of cloud computing resource scheduling, CPU and memory resources are generally used as constraints or the application scheduling virtual machine and the allocation of physical computing resources to the virtual machine are modeled into a constraint satisfaction problem model, which optimizes the allocation of virtual machine resources and improves resource utilization. . Chen Xiaojiao, Chen Shiping and others proposed a group-based multi-objective genetic algorithm virtual machine resource allocation algorithm [7]. Through the improved genetic algorithm, the combination coding and resource requirement coding of the virtual machine are carried out, and the number of physical machines and the physical machine resources occupied by the virtual machines are integrated. The experimental results show that in the process of realizing the matching between physical machines and virtual machines, the algorithm is effective for reducing the number of physical machines used, improving resource utilization, and achieving the purpose of energy saving. But no further research has been done on the correlation between virtual machines.

Seyedeh Aso Tafsiri, Saleh Yousefi studied a combined double auction-based market [8], in which a broker performs the allocation of provider virtual machines according to user requests. The proposed allocation problem is expressed as an integer linear programming model, which aims to maximize the total profit of users and providers. The literature proposed a cloud computing resource optimization allocation strategy based on the game evolution strategy to address the problem of the market resource needs to be allocated on demand and satisfy the rationality and fairness in cloud computing [9]. This strategy uses genetic and evolutionary algorithms to meet the needs of the rationality and fairness of resource allocation from a macro perspective, thus solving the problem that traditional methods only consider individual characteristics.

Because different auction models provide various market-driven resource allocation mechanisms. Literature studies cloud computing resource allocation based on auctions [10]. The article gives an auction-based cloud computing resource allocation framework, discusses the main problems that need to be faced when designing a dynamic resource allocation mechanism, and shares the latest technology of auction-based cloud computing resource allocation.

In solving the problem of dynamic resource allocation, the literature proposed SBS to apply dynamic resource allocation method [11]. When user demand changes dynamically, the application load in SBS will be different at different moments. In order to cope with this change, this method combines the initial static resource allocation scheme with the dynamic resource allocation scheme, and requires the application throughput constraints of SBS. The number of resources required by the application is calculated to calculate the resource allocation time of the application to ensure the end-to-end performance of the SBS application and improve resource utilization.

In a cloud system, due to the continuous changes in requirements and environments, the types and numbers of virtual machines running on nodes need to be continuously adjusted according to requirements. Therefore, Mi Boer and Wang Huaimin proposed a dynamic configuration method of cluster resources [12]. This method is based on the idea of genetic algorithm in the resource allocation under the premise of meeting the needs of users, through the use of chromosome coding, to realize the rapid and dynamic allocation of resources. The Boolean quadratic exponential smoothing method is used to predict user requests, thereby avoiding the reconfiguration result later than the demand change and improving the utilization of cluster resources. However, this method only considers the dynamic allocation of resources within a single cluster. When considering the dynamic allocation of resources across clusters, how to effectively resolve decision conflicts in multiple front-end environments requires further research.

Matching problems are born with the emergence of decision-making, whether in terms of algorithms or in our daily lives, the stability of matching is very important for decision-making [13]. Since David Gale and Lloyd Shapley first proposed the stable matching theory in 1962, the "stability" of matching has received a lot of attention, and research on this topic has also Has been very active. Economists Shapley and Ross Rothy sought a stable solution to the bilateral matching problem from the perspective of mathematics and games, and proposed a game-based stable matching theory research method [13,14]. In 2012, they won the Nobel Economy by virtue of this theory.

3 Stable Matching Resource Allocation Methodology

3.1 Problem Definition

The work goal of cloud resource allocation is to correspond the work requests submitted by users to available resources and to maximize the benefits of the cloud provider while ensuring user performance. As a new business model, from an economic point of view, cloud computing will develop better under the relationship of mutual constraints between users and suppliers. Therefore, this paper considers both cloud users and cloud suppliers when solving the mapping between virtual machines and physical machines.

From the perspective of cloud providers: to minimize the generation of resource fragments on the physical nodes of the cloud data center, thereby improving data center resource utilization efficiency and reducing costs.

From the cloud user's point of view: due to the completion of each user's task is not the same, in order to avoid some users task completion degree is very high and some users task completion degree is zero, need to ensure that the degree of user task completion is relatively fair, so that the user allocated to the resources is relatively fair.

The consideration of user fairness from the user's perspective is mainly reflected in the first phase of task scheduling. Consideration of the vendor's interest is achieved through the VMware Stable Match algorithm proposed in this paper.

The meanings of the letters that appear in this document. u:user, v:virtual machine, m:physical machine, i:task, t:some type of resource, T:number of resource types and, $Rt(m)$ indicates the number of resources available in the physical machine.

3.2 User Fairness

The cloud system schedules tasks based on the priority level of the user making the request. At different moments, as the number of user task requests and the user's task completion level changes, the user's priority will also change. For example, at the beginning, a new user has the highest priority, but at the next moment, when that user requests a new task, his priority drops because he has completed some tasks before. The cloud priority system schedules the highest priority user task which in the queue into runtime, so that may avoid the low priority user task become available, user resource allocation is relatively fair.

User First (UF) is calculated as follows.

$$UF = \mathrm{n_u}/N_u \tag{1}$$

Where *nu* Indicates the number of unfinished tasks in user u. *Nu* Indicates the number of tasks this user needs to complete. The users are sorted in descending order according to the value of UF, the larger the value of UF, the higher the priority of user u. When two users have the same value of UF, they are sorted randomly. When two users have the same *UF* value, they are randomly sorted. In task scheduling, priority is given to scheduling all tasks of the user at the head of the user queue, and this strategy can effectively avoid the situation where users cannot share resources for a long time. This can be seen from Fig. 1.

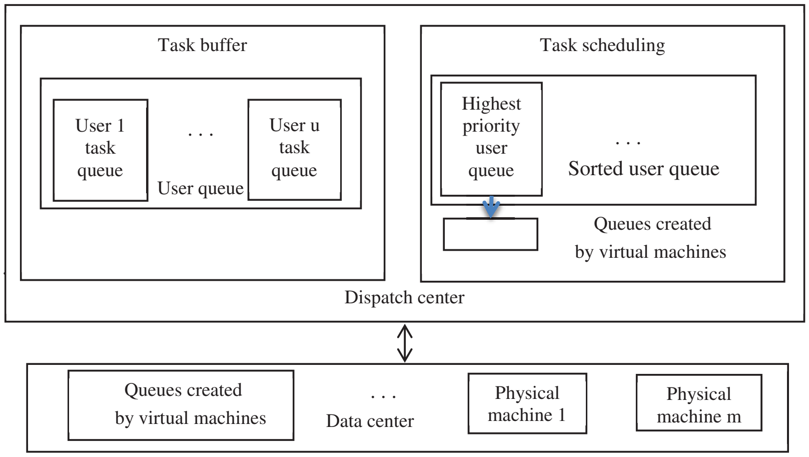

Figure 1: System scheduling diagram

3.4 Vmware Stable Match Algorithm

After analyzing the number of virtual machines to be created, it is necessary to determine which physical machine is more appropriate to place the virtual machine on, and this paper adopts the stable matching theory to achieve stable matching between virtual machines and physical machines. This is a one-to-many matching problem, where one virtual machine can only be placed on one physical machine, and multiple virtual machines can be installed on one physical machine. Due to the variety of types of virtual machines and the variable preference order lists of virtual and physical machines for each other, which change as the algorithm proceeds, traditional stable matching algorithms cannot solve the problem of virtual machine placement. In this paper, a new algorithm, the VMware Stable Match (VSM) algorithm, is proposed using the theory of stable matching.

3.4.1 Preference Relationship Design

The matching sides of the stable placement are the virtual machine to be created and the available physical machine, whose total number is N and M, respectively. The preference of a virtual machine for a physical machine is represented by an NxM matrix VPM, where each row of the matrix is a list of the preferences of a virtual machine for all physical machines, and each element of the matrix stores the number of the physical machine. Similarly, the preference of a physical machine for a virtual machine can be represented by a MxN matrix MPV, where each row of the matrix is a list of the preferences of a physical machine for all virtual machines, and each element of the matrix is the virtual machine number.

Consider first the generation of matrix MP Vs. Physical machines are chosen for virtual machines from the perspective of the cloud provider, with the main consideration being how to maximize resource utilization. The resource proportion deviation ($prdm(vi)$) of a virtual machine *V* (task i) placed on a physical machine is calculated by the following formula.

$$\mathrm{prd_m}(v_i) = \frac{1}{T}\sum_{t=1}^{T}(v_{it}/R_t(m))^2 - [\frac{1}{T}\sum_{t=1}^{T}(v_{it}/R_t(m))]^2 \tag{2}$$

Where *vit* denotes the number of class t resources required to complete task i to create the corresponding virtual machine.

Next consider the generation of the matrix VPM. In calculating the preference list for a virtual machine, two main aspects are considered: the performance of the physical machine and the impact of the user's decision on the choice of the physical machine. The performance of the physical machine is considered in terms of the CPU processing power, the size of RAM and Storage, and the network bandwidth performance of the physical machine. These four factors constitute the condition vector *con(i,m)=(MIPS (m),RAM(m), Sto(m), BAND(m))* of the virtual machine selection, because each task has different weight requirements for these conditions, for example, some tasks need a larger storage space, then it will have a larger proportion of the weight of this aspect, other aspects of the weight of the proportion of relatively small. So the weight of task i for each condition is denoted by the vector *f(i)=(f1(i),f2(i),f3(i),f4(i))*, where the sum of their weights is 1. The user's decision is based on the price of the physical machine and the distance between the physical machine and the user's location to consider its influence on the choice of the virtual machine. For different physical machines, when they can meet the user's needs at the same time, the user will generally choose the cheaper physical machine and the physical machine that is close to them for economic reasons. These two factors constitute the condition vector *ud(i,m)=(price(m), distance(m))* of the virtual machine's choice of physical machine, and since each user has different weight requirements for these two conditions, for example, some users are not bad money, they are more likely to choose the more expensive physical machine to show their identity, then the weight proportion of price will be larger. So the weight of user u on each condition is denoted by the vector *g(i)=(g1(u),g2(u))*, where the sum of their weights is 1. So the preference function for the task can be defined as:

$$pref(i) = con(i, m)f(i) + ud(i, m)g(i) \tag{3}$$

Each physical machine number is sorted from largest to smallest according to the calculated preference function value, and if there are multiple physical machines with the same value, they are randomly sorted to obtain the task preference list.

3.4.2 VMware Stable Match Algorithm

For the issue of stable matching between virtual machines and physical machines, when a willingness to match is established between a physical machine and a virtual machine, but if that virtual machine is to be created, there are not enough resources available on the physical machine. At this point, the cloud provider will need to decide whether to abandon this match or choose to delete some of the virtual machines it has created to make room for the creation of that virtual machine. In order to make this decision, it is first necessary to define a concept, satisfaction level.

Satisfaction: For one physical machine *mi* and a collection of all virtual machines created on that physical machine.

Satisfaction of c(*mi*) is used to measure these members' evaluation of a match, and satisfaction can be defined as:

$$Sat = (m_i, v(m_i)) = \sum v_{ij} \in v(m_i) \frac{Rank(v_{ij}, m_i) + Rank(m_i, v_{ij})}{Num(\mu(m_i))} \tag{4}$$

where $Rank(vij, mi)$ is the rank in the preference list of *mi* each matched *vij* virtual machine *mi*, $Rank(mi, vij)$ is the rank in the preference list of each physical machine *mi* for *vij* each virtual machine it matches, and $Num(\mu(m_i))$ is the m_i total number of matched virtual machines.

The steps of VMware stable match algorithm are presented in Algorithm 1.

Algorithm 1: VMware stable match algorithm

```
Input: Creatable queue and preference list.
Output: Virtual machine's assignment.
Initial: Set all VMs to be free.
Step 1: Each virtual machine processes one user task that has dependencies
Step 2: All the virtual machines in the creatable queue send a request to the physical machine with the
        highest preference in its own preference list.
Step 3: Physical machine selects the one with the highest preference ranking from the virtual machines that
        sent the request to it according to its own preference list.
Step 4: Whether its own available resources can create it? If YES, go to step 5. NO, go to step 6.
Step 5: Removes this virtual machine from the creatable queue, and the physical machine receives the
        virtual machine and rejects the others virtual machine.
Step 6: Whether to abandon this match or replace one or more previously created lower-ranked virtual
        machines with a new virtual machine, based on the level of satisfaction. If the replacement
        procedure is selected, go to step 7, otherwise go to step 8.
Step 7: All replaced VMs are added to the createable queue again and the physical machine is removed from
        its own preference list to move on to the next round of matching.
Step 8: The unaccepted virtual machine is rejoined in the creatable queue, removing that physical machine
        from the preference list and moving on to the next round of matching.
Step 9: Update the existing preference list matrix before the next match.
Step 10: Repeat steps 2 through 9 until all virtual machines are not free.
```

See this algorithm we can know that initially we need to set all VMs to be free. And all the virtual machines in the creatable queue send a request to the physical machine with the highest preference in its own preference list, and the physical machine selects the one with the highest preference ranking from the virtual machines that sent the request to it according to its own preference list, and determines whether its own available resources can create it, and if so, removes this virtual machine from the creatable queue, and the physical machine receives the virtual machine and rejects the others virtual machine. If not, determine whether to abandon this match or replace one or more previously created lower-ranked virtual machines with a new virtual machine, based on the level of satisfaction.

At a decision moment, task scheduling is required to sort U users with time complexity *O(U)*; the time complexity of the virtual machine stability matching algorithm is *O(mn)*, and m, n are the number of physical machines and virtual machines to be created respectively, every physical machine has n virtual machines.

4 Simulation Experiments

Suppose at some point in time, there are users u1, u2, u3 in the cloud system, u1 has two unfinished tasks $u1i1$, $u1i2$ no completed tasks, u2 unfinished tasks $u2i1$, 4 completed tasks, u3 unfinished tasks 2, 2 completed tasks. There are physical machines in the cloud A, B, A, B resources available in Tab. 1, $u1i1$ $u1i2$, $u3i1$, 1 type of resources $u3i2$ needed 2 types of resources, respectively, see Tab. 2, the task of the corresponding virtual machine with v said. As$u1i1$ means that each task is independent, the CPU processing power, RAM and storage sizes, and network bandwidth performance of each physical machine are shown in Tab. 3., (1/3,2/9,1/9,1/3), (2/9,1/9,1/9,5/9). The prices of physical machines A and B are $10,000 and $30,000, respectively, and the distances between physical machines A and B and users u1, u2, u3 are shown in Tab. 4. The weight vectors of users u1, u2, u3 for distance and price are (7/12,5/12), (2/3,1/3), and (5/6,1/6), respectively The following are the steps to achieve a stable match between a virtual machine and a physical machine.

Table 1: Types and quantity of resources available in physical machines

Type of resource Physical machine	Category 1 resources	Category 2 resources
A	20	18
B	15	30

Table 2: Type and level of resources required for mandate implementation

Type of resource Mandate	Category 1 resources	Category 2 resources
$u1_{i1}$	5	7
$u1_{i2}$	3	8
$u2_{i1}$	4	12
$u3_{i1}$	3	4
$u3_{i2}$	2	5

Table 3: CPU processing power, RAM, storage, and network bandwidth performance available for each physical machine

Type of resource Physical machine	CPU processing power (10 being the best)	RAM size	Storage size	Network broadband performance (10 being optimal)
A	6	84 GB	500 GB	7
B	7	32 GB	512 GB	9

Table 4: Distances between individual physical machines and users

User Physical machine	User u1	User u2	User u3
A	150 km	30 km	100 km
B	90 km	60 km	30 km

According to Eq. (1), the priority of user u1 is UF1=2/2=1, the priority of user u2 is UF2=1/5, and the priority of user u3 is UF3=2/4=1/2, 1>1/2>1/5, so the priority of u1, u2, and u3 are u1>u3>u2 respectively. first scheduling user u1, the task , corresponding $u1i1$ to the pending $u1i2$. The resources needed for the virtual machine are $V1U1i1$=(5,7),$V2U1i2$=(3,8) which can be created as queues $V1U1i1$ and $V2U1i2$, and the available resources for physical machines A,B are = (20, 18), = (15, 30), respectively.

The resource ratio deviations between virtual and physical machines are calculated from Eq. (2) as:

$$\mathrm{prd_A}(\mathrm{v1_{u1_{i1}}}) = 0.5 \times \left[(5/20)^2 + (7/18)^2\right] - [0.5 \times (5/20 + 7/18)]^2 = 0.004822$$

$$\mathrm{prd_A}(\mathrm{v2_{u1_{i2}}}) = 0.5 \times \left[(3/20)^2 + (8/18)^2\right] - [0.5 \times (3/20 + 8/18)]^2 = 0.021674$$

$$\mathrm{prd_B}(\mathrm{v2_{u1_{i2}}}) = 0.5 \times \left[(3/15)^2 + (8/30)^2\right] - [0.5 \times (3/15 + 8/30)]^2 = 0.001111$$

$$\mathrm{prd_B}(\mathrm{v1_{u1_{i1}}}) = 0.5 \times \left[(5/15)^2 + (7/30)^2\right] - [0.5 \times (5/15 + 7/30)]^2 = 0.0025$$

Calculated from Eq. (3) we can get:

$\mathrm{pref}_A(\mathrm{u1_{i1}})$=$(5/12, 1/6, 1/4, 1/6) \bullet (6, 84, 500, 7) + (7/12, 5/12) \bullet (1, 150)$=205.75

$\mathrm{pref}_B(\mathrm{u1_{i1}})$=$(5/12, 1/6, 1/4, 1/6) \bullet (7, 32, 512, 9) + (2/3, 1/3) \bullet (3, 90)$=169.5

$\mathrm{pref}_A(\mathrm{u1_{i2}})$=$(2/3, 1/9, 1/9, 1/9) \bullet (6, 84, 500, 7) + (7/12, 5/12) \bullet (1, 150)$=132.75

$\mathrm{pref}_B(\mathrm{u1_{i2}})$=$(2/3, 1/9, 1/9, 1/9) \bullet (7, 32, 512, 9) + (2/3, 1/3) \bullet (3, 90)$=98.111111

So the preference ordering matrices MPV of physical machines A and B for virtual machines v1 and v2 and the initial preference ordering matrices VPM of virtual machines v1 and v2 for physical machines A and B, respectively, are as follows.

$$\begin{bmatrix} \mathrm{v1} & \mathrm{v2} \\ \mathrm{v2} & \mathrm{v1} \end{bmatrix} \begin{bmatrix} A & B \\ A & B \end{bmatrix}$$

MPV VPM

First round match, the queues can be created as v1, v2. v1 sends requests to A and v2 also sends requests to A. The preference list of physical machine A shows that A prefers v1. A's available resources are sufficient to create v1, so A creates v1, rejects v2, removes A from the preference list of v2, and updates the preference list. The new preference list is as follows.

$$\begin{bmatrix} \mathrm{v1} & \mathrm{v2} \\ \mathrm{v2} & \mathrm{v1} \end{bmatrix} \begin{bmatrix} A & B \\ B & \end{bmatrix}$$

MPV VPM

Second round match, the available resources of B are sufficient to create v2, and B creates v2. At this point, all VMs in the queue reach the match, and no VM can break the match to achieve stable matching.

At the next moment, the user priority is recalculated based on the requested user and the user's task completion, e.g., if user 1 has 1 new task request at this moment, then his priority is 1/3, then the priority of the other users is recalculated in the same way, and after arriving at the queue that can be created by the virtual machine, the preference list is calculated, and the mapping between the virtual machine and the physical machine is derived based on the virtual machine stable matching algorithm. Just because the last round of user 1 tasks took up some resources, the resources available to physical machines A and B at this time are different from what they were before.

5 Summary and Outlook

Resource allocation is the assignment of available resources to various uses. In cloud computing management, resource allocation is the scheduling of activities while taking into consideration both the resource availability and the stability. In our strategic planning, resource allocation is a plan for using available resources to achieve goals for the future optimization. It is the process of allocating VM resources among the various physical units.

Our method may be contingency mechanisms. Because a priority ranking of tasks excluded from the virtual machines, showing which tasks become available and a priority ranking of some tasks included in the plan. Our resource allocation method comprehensively consider the needs of cloud users and cloud providers. When multiple users apply for resources, at first, selects a user by user priority, and then deals with this user's task. This action avoids long-term sharing of resources by one user, brings a relatively fair user task scheduling. After analyzing the type and number of virtual machines to be created, the stability matching theory is applied to design the preference relationship between physical machines and virtual machines for each other to achieve stable matching between virtual machines and physical machines.The experiments are clearly explain the improvement on the resource utilization.

Funding Statement: This work was supported in part by the National Natural Science Foundation of China, Grant Number 72073041. Open Foundation for the University Innovation Platform in the Hunan Province, Grant Number 18K103.2011 Collaborative Innovation Center for Development and Utilization of Finance and Economics Big Data Property. Hunan Provincial Key Laboratory of Finance & Economics Big Data Science and Technology. 2020 Hunan Provincial Higher Education Teaching Reform Research Project under Grant HNJG-2020-1130, HNJG-2020-1124. 2020 General Project of Hunan Social Science Fund under Grant 20B16. Scientific Research Project of Education Department of Hunan Province (Grand No. 20K021), Social Science Foundation of Hunan Province (Grant No. 17YBA049).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] R. M. Pandharpatte, “A review: Resource allocation problem in cloud environment,” *International Journal of Engineering and Technology*, vol. 9, no. 3, pp. 1695–1700, 2017.

[2] N. Minh, N. Pham, V. S. Le and H. C. Nguyen, “Energy, efficient resource allocation for virtual services based on heterogeneous shared hosting platforms in cloud computing,” *Cybernetics and Information Technologies*, vol. 17, no. 3, pp. 47–58, 2017.

[3] W. W. Lin and D. Y. Qi, “A review of resource scheduling research in cloud computing,” *Computer Science*, vol. 39, no. 10, pp. 1–6, 2012.

[4] L. Li, W. J. Wang and F. Zhang, “A cloud computing data center-oriented energy consumption modeling approach,” *Journal of Software*, vol. 25, no. 7, pp. 1371–1387, 2014.

[5] Z. Lei, “Virtual machine resource allocation algorithm in cloud computing,” *Computer Modelling & New Technologies*, vol. 18, no. 11, pp. 279–284, 2014.

[6] H. W. Li, “Resource optimization strategy in cloud computing environment,” *Computer Knowledge and Technology*, vol. 9, no. 35, pp. 7929–7930, 2013.

[7] H. Wang, H. Tianfield and Q. Mair, “Auction based resource allocation in cloud computing,” *Multiagent and Grid Systems*, vol. 10, no. 1, pp. 51–66, 2014.

[8] A. B. M. B. Alam, M. Zulkernine and A. Haque, “A reliability-based resource allocation approach for cloud computing,” in *IEEE 7th Int. Sym. on Cloud and Service Computing (ISCSC)*, Kanazawa, Japan, pp. 249–252, 2017.

[9] F. A. Al-Zahrani, I. Khan, M. Zareei, A. Zeb and A. Waheed, “Resource allocation and optimization in device-to-device communication 5g networks,” *Computers, Materials & Continua*, vol. 69, no. 1, pp. 1201–1214, 2021.

[10] P. Baldoss and G. Thangavel, “Optimal resource allocation and quality of service prediction in cloud,” *Computers, Materials & Continua*, vol. 67, no. 1, pp. 253–265, 2021.

[11] J. Han, W. Jiang, J. Shi, S. Xin, J. Peng *et al.,* “A method for assessing the fairness of health resource allocation based on geographical grid,” *Computers, Materials & Continua*, vol. 64, no. 2, pp. 1171–1184, 2020.

[12] J. Zhe, L. Pan and X. Liu, “A novel cloud workflow scheduling algorithm based on stable matching game theory,” *Journal of Supercomputing*, vol. 12, no. 3, pp. 1–28, 2021.

[13] H. Zhu, “Research on maximum return evaluation of human resource allocation based on multi-objective optimization,” *Intelligent Automation & Soft Computing*, vol. 26, no. 4, pp. 741–748, 2020.

[14] Z. Liu, S. Zhang, Y. Liu, X. Wang and D. Yin, “Run-time dynamic resource adjustment for mitigating skew in mapreduce,” *Computer Modeling in Engineering & Sciences*, vol. 126, no. 2, pp. 771–790, 2021.